CWI
Centrum voor Wiskunde en Informatica
REPORT*RAPPORT*

Presuppositions and information updating

D.J.N. van Eijck

Computer Science/Department of Software Technology

CS-R9546 1995

Report CS-R9546
ISSN 0169-118X

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

Presuppositions and Information Updating

Jan van Eijck

CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

and

OTS
Trans 10, 3512 JK Utrecht, The Netherlands

jve@cwi.nl

Abstract

Presupposition failures are errors occurring during the left-right processing of a computer program or natural language text. A general method for analysing such errors with dynamic logic is presented, based on the idea that sequential processing changes context dynamically and that this process of context change can be made the object of analysis in dynamic modal logic.

1991 CR Subject Classification: F.3.1, F.3.2, I.2.4, I.2.7.
Keywords and Phrases: semantics of natural language, presupposition, semantics of programming, error analysis, modal logic, dynamic logic, knowledge representation languages.
Note: to appear in M. Kanazawa, C. Piñón, H. de Swart (eds.), *Quantifiers, Deduction, and Context*, CSLI, Stanford.

1 Introduction

Updating a state of information with a statement bearing a presupposition can be viewed as a combination of two things: (i) checking whether the presupposition holds in the current context, and, provided this is the case, (ii) updating the current state of information with the informational content of the statement. In case the presupposition is not fulfilled in the current context, one might adjust the context to make it hold, and next do the further update with the informational content of the statement. These two actions are obviously *ordered*. First the context is adjusted to accommodate the presupposition, next the information state is updated with the assertion. Context and information state are used interchangeably here, and indeed, we can view the context which gets updated as a state of information.

We propose to view the study of presupposition failure from the general perspective of updating information structures (van Benthem 1991, de Rijke 1994, Jaspars to appear). In this perspective, providing information is a dynamic process involving a speaker and an audience, which gets the audience from an initial information state to a new more informed state, or in case the new information is inconsistent with the current state, to the absurd information state.

It turns out that in many cases the information update relation is functional. In such cases presupposition failure can be catered for by switching to a partial update function. In case the presupposition of some update is not met in some information state, the update function yields 'undefined' for that update in that state. But information updating is not always functional. Communication mismatches do occur if the new information is too vague to yield a unique update in the current information state. In such cases the audience will have to indicate that the update cannot be processed without further ado. Note that this is different from the communication mismatch which occurs if the new information hinges upon a wrong assumption about the current context (the cases of presupposition failure).

The point that presupposition has to do with information updating has been made again and again in the literature. Stalnaker and Karttunen come to mind as early proponents of the view that presupposition projection should be accounted for in dynamic terms. See Stalnaker 1972, Stalnaker 1974, Karttunen 1973, Karttunen 1974, Karttunen and Peters 1979, and Heim 1983. Modern versions of this approach appear in Beaver 1992, van Eijck 1994, Krahmer 1994, and Zeevat 1992.

Stalnaker proposes the following explanation of the rule (first stated in Karttunen 1973) that the presupposition of a conjunction $A\&B$ consists of the presupposition of A conjoined with the implication $\text{ass}_A \rightarrow \text{pres}_B$.

> The explanation goes like this: ...when a speaker says something of the form A *and* B, he may take it for granted that A ...after he has said it. The proposition that A will be added to the background of common assumptions before the speaker asserts that B. Now suppose that B expresses a proposition that would, for some reason, be inappropriate to assert except in a context where A, or something entailed by A, is presupposed. Even if A is *not* presupposed initially, one may still assert A *and* B since by the time one gets to saying that B, the context has shifted, and it is by then presupposed that A.
> From Stalnaker 1974, p. 211, also quoted in Heim 1992.

Modern versions of the dynamic approach to presupposition all attempt to formalize the *pragmatic* notion of presupposition, i.e., the notion of presupposition where a presupposition is a presupposition of the speaker about the context when he/she utters a sentence in that context, rather than a property of the sentence itself. On the other hand, inappropriateness of a sentence in a given context can be construed as lack of a definite truth value of that sentence in that context, and the presupposition of a sentence can always be viewed as a statement which holds in precisely those contexts where the sentence is appropriate.

We should distinguish, therefore between talking about what holds in given states of information on one hand and about updating information states on the other. It is precisely here that dynamic logic, which distinguishes between a level of static description and a level of procedural description, and which provides means of relating these two description levels, becomes an illuminating tool. The difference between the present analysis and other recent dynamic approaches to presupposition is the focus on the *link* between statics and dynamics, which relates semantic presupposition to pragmatic presupposition.

2 Information Structures

An information structure I is a pair $\langle S, \sqsubseteq\rangle$ with S a non-empty set of information states and $\sqsubseteq$ a pre-order (transitive and reflexive, but not necessarily antisymmetric) over S which is called the information order.

If L is a language for S, then an L-information model M is a triple $\langle S, \sqsubseteq, \sigma\rangle$, where $\sigma : L \rightarrow \mathcal{P}S$ is a specification function which interprets the language L in S. Classical languages can be specified by a single function σ, but for languages of partial logic one needs pairs σ^+, σ^- of such functions, and so on.

We consider the simplest case first, the case of propositional logic, where the states of information are sets of propositional valuations. Let a set of proposition letters P be given. Then the set of valuations is the set $\{0,1\}^P$. Call this set W. Members of W may be considered as '(epistemically) possible worlds'. An information state is a subset of W plus a member of W (the distinguished member plays the role of the 'actual world' or the 'current perspective of the knowing subject'); the set S of all information states is $\{\langle i, w\rangle \mid i \subseteq W, w \in W\}$. The information ordering $\sqsubseteq$ on S is given by:

$$\langle i, w\rangle \sqsubseteq \langle j, w'\rangle \text{ iff } i \supseteq j \text{ and } w = w'.$$

Note that this gives a partial order, not just a pre-order. Note also that we do not demand $w \in i$ if $\langle i, w\rangle$ is an information state. In particular, $\langle \emptyset, w\rangle$ is an information state, namely the absurd information state, viewed from perspective w. Without 'perspective worlds' it would be more awkward to modally characterize inconsistent information. With the use of a perspective world, we can simply say that $\Box\bot$ characterizes an inconsistent belief state.

Information states can be viewed as K45 models: sets of possible worlds with an 'almost' universal accessibility relation, i.e., transitive and almost reflexive and almost symmetric, for the 'perspective world' need not be accessible from itself. To be precise, K45 logic is complete for transitive and euclidean frames (a relation R is euclidean if xRy and xRz together imply that yRz),

K45 models are appropriate to talk about the kind of belief where one has complete information about one's own uncertainty (for more information about this, see Moore 1984). An appropriate 'local' language L to talk about information states is the language of propositional modal logic:

$$\alpha \quad ::= \quad \bot \mid p \mid \neg\alpha \mid (\alpha_1 \wedge \alpha_2) \mid \Diamond\alpha.$$

We employ the usual abbreviations for $\top, \wedge, \rightarrow, \leftrightarrow, \Box$. The specification function σ for the language is given by:

$$\begin{array}{lcl} \sigma(\bot) & = & \emptyset, \\ \sigma(p) & = & \{\langle i, w\rangle \in S \mid w(p) = 1\}, \\ \sigma(\neg\alpha) & = & S - \sigma(\alpha), \\ \sigma(\alpha_1 \wedge \alpha_2) & = & \sigma(\alpha_1) \cap \sigma(\alpha_2) \\ \sigma(\Diamond\alpha) & = & \{\langle i, w\rangle \in S \mid \exists w' \in i \text{ with } \langle i, w'\rangle \in \sigma(\alpha)\}. \end{array}$$

We say that $s \models \alpha$ iff $s \in \sigma(\alpha)$. Note that $\langle i, w\rangle \models \Box\bot$ iff $i = \emptyset$. An information state $\langle \emptyset, w\rangle$ is absurd or inconsistent: if we are in such a state, nothing at all is compatible with what we know or believe. Any boxed formula is true in an absurd information state; indeed, $\langle i, w\rangle \neq \langle \emptyset, w\rangle$ iff there is a formula $\alpha \in L$ with $\langle i, w\rangle \not\models \Box\alpha$.

An information state $\langle W, w\rangle$ is an information state of complete ignorance, a state of having no information at all. If P is infinite, there is no formula of L which characterizes W; for finite P there is. If $P = \{p_0, \ldots, p_n\}$, let C be the (finite) set of all conjunctions of form $\Diamond((\neg)p_0 \wedge \cdots \wedge (\neg)p_n)$. Then:

$$\langle i, w\rangle \models \bigwedge C \text{ iff } i = W.$$

3 Updating Propositional Information

Updating a state of information with a new piece of information can be viewed as moving up in the information order, toward some more informed state. Propositional updates are just tests (performed in the current perspective world). Epistemic updates can shift the information state: updating the information of one's audience with $\Box F$ will get the audience in a state where F is known, i.e., a state where $\Box F$ holds. Similarly, downdating with $\Box F$ will get the audience in a state where F is not known anymore, i.e, in a state where $\Diamond\neg F$ holds.

In the more general perspective on information structures, neither updates or downdates need to be minimal (cf. van Benthem 1991 and de Rijke 1994), but for present purposes this restriction is useful. Minimal updates are given by:

$$\begin{array}{lcl} [\![\alpha^u]\!] & = & \{\langle s, s'\rangle \in S \times S \mid s \sqsubseteq s', s' \in \sigma(\alpha), \\ & & \forall s'' \in S : (s \sqsubseteq s'' \sqsubseteq s' \;\&\; s'' \in \sigma(\alpha)) \Rightarrow s' \sqsubseteq s''\} \end{array}$$

We will use F, G, F_1, ... as metavariables for purely propositional formulas of L. As it turns out, the minimal update relation for L is functional for updates of the form $\Box F$ or $\Diamond F$.

We may assume that knowing subjects, unless they are all-knowing, cannot distinguish the actual world from any other of their epistemic alternatives, so updates with purely propositional F are a bit silly: they can succeed only if they do not change states, and their success depends on what is true in the actual world.

Example 1 An update with $p \vee q$ in state $\langle i, w\rangle$ checks if $i, w \models p \vee q$, and succeeds if this is the case, fails otherwise. In other words, this update succeeds iff $w(p) = 1$ or $w(q) = 1$.

If I utter $p \vee q$, this should be understood as: 'I believe or know that $p \vee q$, and I want you to accept that information about the world, too. So, this update should be understood as having an implicit $\Box$ in front.

Example 2 An update with $\Box(p \vee q)$ in state $\langle i, w\rangle$ causes a shift to a state $\langle j, w\rangle$ where $j = \{w' \in i \mid \langle i, w'\rangle \models p \vee q\}$.

Example 3 An update with $\Diamond(p \vee q)$ in state $\langle i, w\rangle$ does not change state in case $\langle i, w\rangle \models \Diamond(p \vee q)$, and otherwise fails.

Updates with $\Diamond F$ formulas are 'consistency tests': they check whether the information F is consistent with the current information state. Let $\|\alpha\|_i$ be $\{w \in W \mid \langle i, w\rangle \in \sigma(\alpha)\}$. The following proposition holds:

Proposition 1

1. $[\![F^u]\!] = \{\langle s, s\rangle \in S \times S \mid s \models F\}$.
2. $[\![\Box F^u]\!] = \{\langle\langle i, w\rangle, \langle i \cap \|F\|_i, w\rangle\rangle \mid \langle i, w\rangle \in S\}$.
3. $[\![\Diamond F^u]\!] = \{\langle s, s\rangle \in S \times S \mid s \models \Diamond F\}$.

Example 4 An update with $\Box p \vee \Box q$ in state $\langle i, w\rangle$ is not functional in case i contains a w_1 with $w_1(p) = 1, w_1(q) = 0$ and a w_2 with $w_2(p) = 0, w_2(q) = 1$. In this case there are two possible outcome states: $s_1 = \langle\{w' \in i \mid w'(p) = 1\}, w\rangle$ and $s_2 = \langle\{w' \in i \mid w'(q) = 1\}, w\rangle$.

Take for example the case where $s = \langle\{p\bar{q}, \bar{p}q\}, w\rangle$ (where $\bar{p}$ indicates that p is false). Both

$$\langle\{p\bar{q}, \bar{p}q\}, w\rangle \mapsto \langle\{p\bar{q}\}, w\rangle$$

and

$$\langle\{p\bar{q}, \bar{p}q\}, w\rangle \mapsto \langle\{\bar{p}q\}, w\rangle$$

are minimal updates.

Modal updates are discussed in Veltman 1991, but with the constraint that only modal updates of the forms $\Diamond\top \rightarrow \Diamond F$ and $\Box F$ are allowed. Our $\Diamond\top \rightarrow \Diamond F$ corresponds to Veltman's *might* F, and our $\Box F$ to his F, so the $\Box$ is left implicit in his notation. Updates of the form $\Diamond\top \rightarrow \Diamond F$ are total functions, while updates of the form $\Diamond F$ may be partial. In fact, an update with $\Diamond\top \rightarrow \Diamond F$ will effect a transition to an inconsistent state if $\Diamond F$ does not hold in the current state. Any update of the form $\Box F \wedge \bigvee \bigwedge \Diamond F_i$ (where F and all the F_i are purely propositional), is functional. An important result about K45, by the way, is that every formula has an equivalent formula of the form $\bigvee(F \wedge \Box G \wedge \bigwedge \Diamond H_i)$ (where F, G and the H_i are all purely propositional). This follows from the completeness of K45 with respect to finite 'balloon' frames, i.e., frames where the accessibility relation is transitive and euclidean; these frames have the shape of a balloon of mutually accessible worlds all accessible from a single perspective world. See e.g. Chellas 1980 for more information. Disjunction over $\Box$ is the feature that 'threatens' functionality.

The functional K45 formulas are 'honest' formulas in the sense of Halpern and Moses 1985. A formula is honest if one can honestly claim that one *only* knows that formula. This is equivalent to saying that a minimal update of the state of complete ignorance with that formula is functional. Claiming that you only know $\Box p \vee \Box q$ is a cheat, because in order for that to be a true statement you either have to be in a state where all the accessible worlds are p worlds, and in that case you also know p, or you have to be in a state where all the accessible worlds are q worlds, and in that case you also know q.

A K45 formula α is persistent if $\langle i, w\rangle \models \alpha$ and $\langle i, w\rangle \sqsubseteq \langle j, w\rangle$ together imply that $\langle j, w\rangle \models \alpha$. Formulas of the form $\Box F$ are persistent, formulas of the form $\Diamond F$, with F a consistent propositional formula, are not. Conjunctions and disjunctions of persistent formulas are persistent. Every persistent formula is K45 equivalent to a disjunction of conjunctions of formulas of the forms F and $\Box F$ (F purely propositional).

Example 5 $\Diamond p$ is true at $\langle W, w\rangle$, but false at any information state $\langle i, w\rangle$ without p worlds.

The persistent and functional K45 formulas are precisely the formulas of the form $F_1 \wedge \Box F_2$, F_1 and F_2 purely propositional (up to K45 equivalence).

Conversely, we can look at minimal downdates to move back to a state where α does not hold anymore. Minimal downdates are given by:

$$\begin{array}{rcl} [\![\alpha^d]\!] & = & \{\langle s, s'\rangle \in S \times S \mid s' \sqsubseteq s, s' \notin \sigma(\alpha), \\ & & \forall s'' \in S : (s' \sqsubseteq s'' \sqsubseteq s \ \&\ s'' \notin \sigma(\alpha)) \Rightarrow s'' \sqsubseteq s'\}. \end{array}$$

The minimal downdate relation is not functional for formulas of the form $\Box F$, and it is a test for formulas of the form $\Diamond F$.

Example 6 $[\![(\Box p)^d]\!]$ will relate a state $\langle i, w\rangle$ satisfying $\langle i, w\rangle \models \Box p$ to any state $\langle j, w\rangle$ where j is of the form $i \cup \{w'\}$, with $w'(p) = 0$, and a state $\langle i, w\rangle$ not satisfying $\langle i, w, \rangle \models \Box p$ to itself.

Downdates with formulas of the form $\Diamond F$ can only succeed in case $\Diamond F$ is inconsistent with the current information state.

Example 7 $[\![(\Diamond p)^d]\!] = \{\langle s, s\rangle \in S \times S \mid s \models \neg\Diamond p\}$.

In general we have:

Proposition 2

1. $[\![F^d]\!] = \{\langle s, s\rangle \in S \times S \mid s \models \neg F\}$.
2. $[\![\Box F^d]\!] = \{\langle s, s\rangle \in S \times S \mid s \models \neg\Box F\} \cup \{\langle\langle i, w\rangle, \langle i \cup \{w'\}, w\rangle\rangle \mid \langle i, w\rangle \in S, \langle i, w\rangle \models \Box F, w' \in W - ||F||_i\}$.
3. $[\![\Diamond F^d]\!] = \{\langle s, s\rangle \in S \times S \mid s \models \neg\Diamond F\}$.

Over the local language L we now layer a global language L_1, which is the language of the information structure S. L has a procedural and a propositional level; the procedures are (minimal) updating, (minimal) downdating, testing, plus sequential compositions of those, the propositions of L_1 are built from the formulas of L (which act as atoms of L_1) using boolean combination and procedure projections.

$$\begin{array}{lcl} \alpha & ::= & \bot \mid p \mid \neg\alpha \mid (\alpha_1 \wedge \alpha_2) \mid \Diamond\alpha \\ \pi & ::= & \alpha^u \mid \alpha^d \mid \varphi? \mid (\pi_1; \pi_2) \\ \varphi & ::= & \alpha \mid \neg\varphi \mid (\varphi_1 \wedge \varphi_2) \mid \Diamond\varphi \mid \mathrm{dom}\,(\pi) \mid \mathrm{ran}\,(\pi) \mid \mathrm{fix}\,(\pi). \end{array}$$

The interpretation of L_1 consists of two parts: a relational interpretation for the procedures and a truth definition for the formulas. The interpretation for the procedures and formulas uses mutual recursion.

$$\begin{array}{lcl} [\![\alpha^u]\!] & = & \text{as given above.} \\ [\![\alpha^d]\!] & = & \text{as given above.} \\ [\![\varphi?]\!] & = & \{\langle s, s\rangle \in S \times S \mid S, s \models \varphi\}. \\ [\![\pi_1; \pi_2]\!] & = & [\![\pi_1]\!] \circ [\![\pi_2]\!]. \end{array}$$

In the final clause, $\circ$ denotes relational composition.

$$\begin{array}{lll} S, s \models \alpha & \text{iff} & s \models \alpha \\ S, s \models \neg\varphi & \text{iff} & S, s \not\models \varphi \\ S, s \models (\varphi_1 \wedge \varphi_2) & \text{iff} & S, s \models \varphi_1 \text{ and } S, s \models \varphi_2 \\ S, s \models \Diamond\varphi & \text{iff} & s = \langle i, w\rangle \text{ and} \\ & & \text{there is some } w' \in i \text{ with } S, \langle i, w'\rangle \models \varphi \\ S, s \models \mathrm{dom}\,(\pi) & \text{iff} & \exists s' \in S : s[\![\pi]\!]s' \\ S, s \models \mathrm{ran}\,(\pi) & \text{iff} & \exists s' \in S : s'[\![\pi]\!]s \\ S, s \models \mathrm{fix}\,(\pi) & \text{iff} & s[\![\pi]\!]s. \end{array}$$

This system is an extension of the system of update logic presented in Veltman 1991 with tests, downdates and a more liberal regime concerning modal updates. Alternatively, it can be viewed as a fragment of a structured version of the Dynamic Modal Logic (DML) in van Benthem 1991, with structured states (in this case: K45 models) instead of unstructured propositional valuations, but with just a subset of the procedural repertoire.

We can define the perhaps more familiar dynamic logic style procedure modalities $\langle\pi\rangle$ and $[\pi]$ in terms of the projection operators and tests as follows:

$$\langle\pi\rangle\varphi \stackrel{\text{def}}{=} \mathrm{dom}\,(\pi; \varphi?),$$

$$[\pi]\varphi \stackrel{\text{def}}{=} \neg\mathrm{dom}\,(\pi; (\neg\varphi)?).$$

Note that it follows from these definitions that:

- $S, s \models \langle\pi\rangle\varphi$ iff $\exists s' \in S : s[\![\pi]\!]s'$ and $S, s' \models \varphi$.
- $S, s \models [\pi]\varphi$ iff $\forall s' \in S : s[\![\pi]\!]s'$ implies $S, s' \models \varphi$.

Note the following important differences:

$$\begin{array}{lll} S, s \models \bot & & \text{never.} \\ S, s \models \top & & \text{always.} \\ S, s \models \Box\bot & \text{iff} & s = \langle\emptyset, w\rangle. \\ S, s \models \Diamond\top & \text{iff} & s \neq \langle\emptyset, w\rangle. \end{array}$$

Example 8 We see from the above that $[\pi]\bot$ expresses that no π transition is possible, while $[\pi]\Box\bot$ expresses that the only possible π transition will get one to the absurd information state, or, in other words, that the transition π yields inconsistency.

Example 9 $\langle\pi\rangle\top$ expresses that a π transition is possible; $\langle\pi\rangle\Diamond\top$ expresses that a *consistent* π transition is possible.

The notion of validity for L_1 is as follows:

- $\models \varphi$ iff for all $s \in S : S, s \models \varphi$.

Because we have used the full space $\mathcal{P}W$ to define the state set S, there is no need to mention S as a parameter in the validity notion: once the set of proposition letters is fixed, the set of information states is fixed. This changes when we allow S to be a proper subset of

$$\{\langle i, w\rangle \mid i \in \mathcal{P}W, w \in W\},$$

subject to certain conditions. Nothing in the notion of 'information structure' prevents us from doing this, as long as we make sure that the information ordering $\sqsubseteq$ on S remains a pre-order. We will not explore this possibility here, however.

4 Some Example Validities

We will not present a full axiomatisation of the logic of propositional up- and downdating, but merely give some of the valid principles that we can use to reason about information transitions. Next to the obvious axioms and rules of inference of propositional logic, of normal modal logic for $\Diamond$, we need the following:

P 4.1 $\Box\varphi \to \Box\Box\varphi$.

P 4.2 $\Diamond\varphi \to \Box\Diamond\varphi$.

Principles 4.1 and 4.2 are the principles of positive and negative introspection of $K45$ for $\Diamond$.

P 4.3 $\text{dom}\,(\pi) \leftrightarrow \text{dom}\,(\pi; \top?)$.

P 4.4 $\text{ran}\,(\pi) \leftrightarrow \text{ran}\,(\top?; \pi)$.

These are to make sure that we can always assume dom arguments to be of the general form $\pi; \varphi?$, and ran arguments to be of the general form $\varphi?; \pi$.

P 4.5 $[\pi](\varphi_1 \to \varphi_2) \to ([\pi]\varphi_1 \to [\pi]\varphi_2)$.

This is the K schema for π, which expresses that for every information transition procedure π, the operator $[\pi]$ is a normal modal operator.

P 4.6 $\langle\alpha^u\rangle\top \leftrightarrow \langle\alpha^u\rangle\alpha$.

This expresses that after updating with α, α will hold. The principle does not entail that updates have the property of *right seriality* (for every s there is a t with $s[\![\pi]\!]t$), for we have seen that this need not be the case for α of the forms F or $\Diamond F$.

P 4.7 $[F^u]\varphi \leftrightarrow (F \rightarrow \varphi)$.

P 4.8 $[\Diamond F^u]\varphi \leftrightarrow (\Diamond F \rightarrow \varphi)$.

These express that F and $\Diamond F$ updates are tests.

P 4.9 $\langle \Box F^u \rangle \Box G \leftrightarrow [\Box F^u]\Box G \leftrightarrow \Box(F \rightarrow G)$.

The soundness of Principle 4.9 follows from the functionality of $\Box F$ updates, plus the next proposition.

Proposition 3 $s \models [\Box F^u]\Box G$ *iff* $s \models \Box(F \rightarrow G)$.

Proof. $\langle i, w\rangle \models [\Box F^u]\Box G$
iff $\langle i \cap ||F||_i, w\rangle \models \Box G$
iff $\langle i \cap \{w' \mid w' \models F\}, w\rangle \models \Box G$
iff $\forall w' \in i$: if $w' \models F$ then $w' \models G$
iff $\langle i, w\rangle \models \Box(F \rightarrow G)$. ∎

P 4.10 $\langle \Box F^u \rangle \Diamond G \leftrightarrow [\Box F^u]\Diamond G \leftrightarrow \Diamond(F \wedge G)$.

The soundness of Principle 4.10 follows from the functionality of $\Box F$ updates, plus the next proposition.

Proposition 4 $s \models [\Box F^u]\Diamond G$ *iff* $s \models \Diamond(F \wedge G)$.

Proof. $\langle i, w\rangle \models [\Box F^u]\Diamond G$
iff $\langle i \cap ||F||_i, w\rangle \models \Diamond G$
iff $\langle i \cap \{w' \mid w' \models F\}, w\rangle \models \Diamond G$
iff $\exists w' \in i$: $w' \models F$ and $w' \models G$
iff $\langle i, w\rangle \models \Diamond(F \wedge G)$. ∎

P 4.11 $[F^d]\varphi \leftrightarrow (F \vee \varphi)$.

P 4.12 $[\Diamond F^d]\varphi \leftrightarrow (\Diamond F \vee \varphi)$.

These express that F and $\Diamond F$ downdates are tests.

The following principle is a rule rather than an axiom schema.

P 4.13 $\frac{(\bigwedge C \rightarrow \Diamond(F \wedge \neg G))}{(\text{ran }(\Diamond F?; \Box G^u) \leftrightarrow \Box G)}$.

Here $\bigwedge C$ is the conjunction of all formulas of the form $\Diamond((\neg)p_0 \wedge \cdots \wedge (\neg)p_n)$, where $p_0, \ldots, p_n$ are the proposition letters occurring in F, G.

P 4.14 $\text{ran }(\varphi?; F^u) \leftrightarrow (\varphi \wedge F)$.

P 4.15 $\text{ran }(\varphi?; \Diamond F^u) \leftrightarrow (\varphi \wedge \Diamond F)$.

P 4.16 $\text{ran }(\Box F?; \Box G^u) \leftrightarrow \Box(F \wedge G)$.

After updating a context with a persistent update, the persistent preconditions will hold in the new context.

P 4.17 $\text{ran }(\varphi?; F^d) \leftrightarrow (\varphi \wedge \neg F)$.

P 4.18 $\operatorname{ran}(\varphi?; \Diamond F^d) \leftrightarrow (\varphi \wedge \neg\Diamond F)$.

P 4.19 $\operatorname{ran}((\Diamond F_1 \wedge \cdots \wedge \Diamond F_n)?; \Box G^d) \leftrightarrow (\Diamond F_1 \wedge \cdots \wedge \Diamond F_n \wedge \Diamond\neg G)$.

The counterparts to the previous three for downdates.

P 4.20 $\langle \alpha^d \rangle \top \leftrightarrow \langle \alpha^d \rangle \neg\alpha$.

This expresses that a downdate can only succeed if after downdating with α, α does not hold anymore. Note that downdating will be impossible if the downdate is a logical validity (the present set-up is unsuitable for modelling 'unlearning' of logical truths).

P 4.21 $\operatorname{fix}(\alpha^u) \leftrightarrow \alpha$.

This expresses that updating with α doesn't change the context precisely when α already holds.

P 4.22 $\operatorname{fix}(\alpha^d) \leftrightarrow \neg\alpha$.

This expresses that downdates with information that is already known to be false have no effect.

P 4.23 $\operatorname{fix}(\pi; \varphi?) \leftrightarrow \operatorname{fix}(\varphi?; \pi) \leftrightarrow (\operatorname{fix}(\pi) \wedge \varphi)$.

This is an obvious statement about fixpoints.

P 4.24 $(\operatorname{fix}(\pi) \wedge \varphi) \rightarrow (\operatorname{dom}(\pi; \varphi?) \wedge \operatorname{ran}(\varphi?; \pi))$.

This relates fixpoint to domain and range.

P 4.25 $(\operatorname{fix}(\pi_1) \wedge \operatorname{fix}(\pi_2)) \rightarrow \operatorname{fix}(\pi_1; \pi_2)$.

If s is a fixpoint for π_1 and π_2, then s is a fixpoint for $\pi_1; \pi_2$ (but note that this cannot be strengthened to an equivalence).

P 4.26 $\operatorname{dom}(\pi_1; \pi_2; \varphi?) \leftrightarrow \operatorname{dom}(\pi_1; \operatorname{dom}(\pi_2; \varphi?)?)$.

This is the usual principle for sequential composition. Stated in terms of $\langle \pi \rangle$ it can also be expressed as $\langle \pi_1; \pi_2 \rangle \varphi \leftrightarrow \langle \pi_1 \rangle \langle \pi_2 \rangle \varphi$ (familiar from Pratt-style propositional dynamic logic).

P 4.27 $\operatorname{ran}(\varphi?; \pi_1; \pi_2) \leftrightarrow \operatorname{ran}(\operatorname{ran}(\varphi?; \pi_1)?; \pi_2)$.

This is the counterpart to the previous principle. Finally, here are three axioms about testing.

P 4.28 $\operatorname{dom}(\varphi_1?; \varphi_2?) \leftrightarrow (\varphi_1 \wedge \varphi_2)$.

P 4.29 $\operatorname{ran}(\varphi_1?; \varphi_2?) \leftrightarrow (\varphi_1 \wedge \varphi_2)$.

P 4.30 $\operatorname{fix}(\varphi?) \leftrightarrow \varphi$.

It should be noted that some of these principles can be simplified if we extend the relational repertoire of the language. For example, if we admit procedure intersection then we can define fixpoints by means of $\operatorname{fix}(\pi) \leftrightarrow \operatorname{dom}(\pi \cap \top?)$. Of course, the trade-off is that now extra principles for intersection have to be added.

5 Validity and Consequence

While there is an obvious static validity notion for the logic of propositional information transitions (see above), there are several candidates for the notion of 'dynamic valitity' (van Benthem 1991, Veltman 1991, van Eijck and de Vries 1995), which are all easily expressible in the present format.

An information transition π is always accepted if for every information state s, $\langle s, s\rangle \in [\![\pi]\!]$. This is the case iff for every information state s, $s \models \text{fix}\,(\pi)$.

An information transition π is always acceptable if for every information state $s \neq \langle \emptyset, w\rangle$, there is some $s' \neq \langle \emptyset, w\rangle$ with $\langle s, s'\rangle \in [\![\pi]\!]$. This is the case iff for every information state s, $s \models \Diamond\top \rightarrow \langle\pi\rangle\Diamond\top$.

Similarly, while it is clear what the 'static' notion of logical consequence should be (namely, $\Gamma \models \Delta$ iff for every $s \in S$ with $s \models \bigwedge\Gamma$ it holds that $s \models \bigvee\Delta$), there are several candidates for 'dynamic consequence' in this framework (van Benthem 1991; see Kanazawa 1994 for further analysis):

- $\pi_1 \models_1 \pi_2$ iff for all $s \in S$, $s[\![\pi_1]\!]s$ implies $s[\![\pi_2]\!]s$.
- $\pi_1 \models_2 \pi_2$ iff for all $s, s' \in S$ with $s[\![\pi_1]\!]s'$ and $s' \neq \langle\emptyset, w\rangle$ there is an $s'' \neq \langle\emptyset, w\rangle$ with $s'[\![\pi_2]\!]s''$.
- $\pi_1 \models_3 \pi_2$ iff for all $s, s' \in S$, $s[\![\pi_1]\!]s'$ implies $s'[\![\pi_2]\!]s'$.

These are readily expressed in terms of static validity, for we have:

- $\pi_1 \models_1 \pi_2$ iff $\models \text{fix}\,(\pi_1) \rightarrow \text{fix}\,(\pi_2)$.

And for the second one:

- $\pi_1 \models_2 \pi_2$ iff $\models [\pi_1](\Diamond\top \rightarrow \langle\pi_2\rangle\Diamond\top)$.

And the third one:

- $\pi_1 \models_3 \pi_2$ iff $\models \text{ran}\,(\pi_1) \rightarrow \text{fix}\,(\pi_2)$.

6 Information Conveyed by an Update

One way of 'measuring' the information conveyed by an information transition in a state satisfying φ is by means of $\text{ran}\,(\varphi?; \pi)$.

Example 10 The information conveyed by the update $(\Diamond p)^u; (\Box\neg p)^u$ (one of Veltman's key examples) in the state of complete ignorance is calculated as follows:

$$\begin{array}{rcl} \text{ran}\,(\top?; (\Diamond p)^u; (\Box\neg p)^u) & \leftrightarrow & \text{ran}\,(\text{ran}\,(\top?; (\Diamond p)^u)?; (\Box\neg p)^u) \\ & \leftrightarrow & \text{ran}\,(\Diamond p?; \Box\neg p^u) \\ & \leftrightarrow & \Box\neg p. \end{array}$$

The second step uses Principle 4.15, the third Principle 4.13.

Example 11 The information conveyed by the update $(\Box\neg p)^u; (\Diamond p)^u$ in the state of complete ignorance is calculated as follows:

$$\begin{array}{rcl} \text{ran}\,(\top?; (\Box\neg p)^u; (\Diamond p)^u) & \leftrightarrow & \text{ran}\,(\text{ran}\,(\top?; (\Box\neg p)^u)?; (\Diamond p)^u) \\ & \leftrightarrow & \text{ran}\,(\text{ran}\,(\Box\top?; (\Box\neg p)^u)?; (\Diamond p)^u) \\ & \leftrightarrow & \text{ran}\,((\Box\neg p)?; (\Diamond p)^u) \\ & \leftrightarrow & \Box\neg p \wedge \Diamond p. \end{array}$$

This is K45 equivalent to $\bot$, which shows that this update will never succeed. (Note the use of Principle 4.16 in the third step.)

In fact, since updating with Veltman's *might* p corresponds to updating with $\Diamond\top \rightarrow \Diamond p$, the 'rational reconstruction' of Veltman's example is slightly different:

Example 12 The information conveyed by the update

$$(\Box\neg p)^u; (\Diamond\top \rightarrow \Diamond p)^u$$

in the state of complete ignorance is calculated as follows:

$$\begin{array}{l} \text{ran}\,(\top?; (\Box\neg p)^u; (\Diamond\top \to \Diamond p)^u) \\ \leftrightarrow \text{ran}\,(\text{ran}\,(\top?; (\Box\neg p)^u)?; (\Diamond\top \to \Diamond p)^u) \\ \leftrightarrow \text{ran}\,(\text{ran}\,(\Box\top?; (\Box\neg p)^u)?; (\Diamond\top \to \Diamond p)^u) \\ \leftrightarrow \text{ran}\,((\Box\neg p)?; (\Diamond\top \to \Diamond p)^u) \\ \leftrightarrow \Box\neg p \wedge (\Diamond\top \to \Diamond p). \end{array}$$

This is K45 equivalent to $\Box\bot$, which shows that this update will get the audience in an inconsistent state of information.

To see that the final step in the calculation is correct, note that the update procedure $(\Diamond\top \to \Diamond p)^u$ is equivalent to the procedure $(\Diamond\top?; \Diamond p^u) \cup \Box\bot^u$, where $\cup$ denotes choice between procedures. An obvious principle governing choice is:

$$\text{ran}\,(\pi_1; (\pi_2 \cup \pi_3)) \leftrightarrow (\text{ran}\,(\pi_1; \pi_2) \vee \text{ran}\,(\pi_1; \pi_3)).$$

Using this and the other principles, the final step can easily be validated.

In general, a transition π is consistent in state s iff $\langle\pi\rangle\Diamond\top?$ holds in s. This expresses that a transition from s via π is possible which does end up in a consistent state of information.

It is tempting, especially in the light of the dynamic consequence notion $\models_1$, to equate the information conveyed by an information transition π with fix (π). But note that the combination $\Diamond p^u; \Box\neg p^u$ does not have a fixpoint, while, as we have just seen, updating with that information starting from complete ignorance yields a *consistent* information state. Is there perhaps something funny about updates of the form $\Diamond F^u$?

From the present perspective, such updates are indeed strange. Recall that our intention is to model the knowledge of an *audience* addressed by a single speaker. If one assumes that $\Diamond F^u$ corresponds to an assertion by the speaker, then the update would have to correspond to an assertion about the state of knowledge of the audience: the speaker states that F is consistent with what the audience knows already. This assertion would correspond to something like 'I take it that you know that F is possible'. But this is not an assertion in the sense of 'statement influencing the state of knowledge of the audience'.

Compare this with the 'updates' of the form 'p may be the case', or 'maybe p' in Veltman 1991. Veltman renders the assertion 'maybe p' as an update with $\Diamond\top \to \Diamond p$. The big difference between Veltman's information states and ours is that Veltman's information states model the knowledge of a single agent reporting on how he or she processes incoming information, while ours model the knowledge of the *audience* addressed by a single speaker.

In our set-up, updates of the form $\Diamond\top \to \Diamond p$ are not consistency checks of one's own knowledge, as they are for Veltman, but statements about the knowledge of the audience. Since we cannot in general assume that a speaker has complete knowledge of what his or her audience believes, such statements are rather pointless. On the other hand, checking the knowledge of the audience by means of a test $\varphi?$ may still make eminent sense, as we will see in the next section.

In the present set-up, an assertion of the form 'maybe p' should be construed as an invitation to the audience to *reconsider* the truth of p, i.e., such an assertion is a *downdate*, and it has the form $(\neg\Diamond p)^d$, or equivalently $(\Box\neg p)^d$.

If one imposes the constraint that information transitions always be compositions of basic units of the forms $\Box F^u$ and $\Box F^d$, possibly interspersed with tests, then information content can always be described in terms of fixpoints. A consistent fixpoint for $\Diamond p^u; \Box\neg p^u$ does not exist, but for $\Box\neg p^d; \Box\neg p^u$ it does; indeed, any state where $\Box\neg p$ holds is such a fixpoint.

7 Expressing Presuppositions

We have seen that realistic information transitions in our set-up have the forms $(\Box F)^u$ or $(\Box F)^d$. In case such an information transition has a presupposition, we may assume that this has the form of a test to see whether something is known in the current context, i.e., a test of the form $\Box\varphi?$ Not only updates may have presuppositions, witness (1).

(1) Maybe the king of France is eating frog legs.

If I assert (1) then I presuppose that the king of France exists, and I invite my audience to revise their belief that his majesty is doing something other than eating frog legs. (We have seen that 'maybe statements' turn up as downdates of the form $\Box F^d$ in the present framework.) So this is a downdate with presupposition.

The framework presented above has all the machinery in place to express presuppositions. We can express the requirement that updating with α has presupposition φ in state s by means of the complex update $\varphi?;\alpha^u$.

Example 13 $\Box p?;\Box q^u$ succeeds in state s iff $s \models \Box p$, and effects a transition to a state s' with $s \sqsubseteq s'$ and $s' \models \Box q$. (And of course also $s' \models \Box p$, for $\Box p$ is a persistent formula.)

The semantic clause for $[\![\varphi?;\alpha^u]\!]$ is given by:

$$\begin{aligned} [\![\varphi?;\alpha^u]\!] &= [\![\varphi?]\!] \circ [\![\alpha^u]\!] \\ &= \{\langle s, s'\rangle \in [\![\alpha^u]\!] \mid s \models \varphi\} \\ &= [\![\alpha^u]\!] - \{\langle s, s'\rangle \in S \times S \mid s \not\models \varphi\}. \end{aligned}$$

As the relational interpretation demonstrates, $\varphi?;\alpha^u$ is interpreted as an update with α under the presupposition that φ holds in the current context. Similarly, $\varphi?;\alpha^d$ is interpreted as a downdate with α under the presupposition that φ holds in the current context.

Calculating the presupposition of an information transition π consists in finding a specification of the information states s for which there is an s' with $\langle s, s'\rangle \in [\![\pi]\!]$. In these cases we say that the transition π *does not abort*. Conversely, the presupposition failure conditions of an information transition π consist of a specification of the information states s for which there is no s' with $\langle s, s'\rangle \in [\![\pi]\!]$. We say in these cases that transition π *aborts* in state s.

To calculate the presupposition of an update α^u, we have to check the conditions on states s under which the relation $[\![\alpha^u]\!]$ does have a successor for s. These are given by the following schemata, which are derivable from the principles in the previous section:

T 7.1 $\langle\varphi?;\alpha^u\rangle\top \leftrightarrow \varphi \wedge \langle\alpha^u\rangle\top$.

This expresses that a simplex update with presupposition can be performed if and only if the presupposition holds in the current information state and the update without presupposition is possible in the current context.

T 7.2 $\langle\varphi?;\alpha^d\rangle\top \leftrightarrow \varphi \wedge \langle\alpha^d\rangle\top$.

This expresses that a downdate under presupposition is possible iff the presupposition holds in the current information state and the downdate without presupposition is possible in that state. Note that the presupposition of an information transition is nothing but the weakest preconditions for success of that transition, in the well known computer science sense.

For a concrete example, assume that the lexical presupposition of being a bachelor consists of being male plus being adult. We do not yet look inside the basic propositions built from these predicates, so we merely say that example (2) presupposes the conjunction of (3) and (4), and asserts (5).

(2) Jan is a bachelor.

(3) Jan is male.

(4) Jan is adult.

(5) Jan is unmarried.

Basic propositions that do not themselves have presuppositions can be represented using basic proposition letters. Let us use p for (3), q for (4), and $\neg r$ for (5).

The update for *Jan is a bachelor* does have the presuppositions *Jan is male* and *Jan is adult*, and (after the update with these presuppositions) it makes the assertion *Jan is unmarried*, so it can be represented as $\Box(p \wedge q)?; \Box\neg r^u$. Let P be the set $\{p, q, r\}$. Information states for this fragment are built from valuations in $\{p, q, r\} \to \{0, 1\}$.

(6) Jan is male. Jan is a bachelor.

The meaning of the update with the sequence (6) is given by $[\![\Box p^u; \Box(p \wedge q)?; \Box\neg r^u]\!]$. We write this out to check its meaning:

$$\begin{array}{l} s[\![\Box p^u; \Box(p \wedge q)?; \Box\neg r^u]\!]s' \\ \text{ iff } \exists s'' : s[\![\Box p^u]\!]s'' \text{ and } s''[\![\Box(p \wedge q)?; \Box\neg r^u]\!]s' \\ \text{ iff } \exists s'' : s[\![\Box p^u]\!]s'' \text{ and } s'' \models \Box(p \wedge q) \text{ and } s''[\![\Box\neg r^u]\!]s' \\ \text{ iff } \exists s'' : s[\![\Box p^u]\!]s'' \text{ and } s'' \models \Box q \text{ and } s''[\![\Box\neg r^u]\!]s'. \end{array}$$

It follows from this that the non-abort condition on s is given by:

$$\exists s' : s[\![\Box p^u]\!]s' \text{ and } s' \models \Box q.$$

This is the case iff $s \models \Box(p \to q)$. In other words, the presupposition is that it is known in the current context that if Jan is male then he is adult. We can also derive this in the calculus, as follows:

$$\begin{array}{rcl} \langle \Box p^u; \Box(p \wedge q)?; \Box\neg r^u\rangle\top & \leftrightarrow & \langle \Box p^u\rangle\langle \Box(p \wedge q)?\rangle\langle \Box\neg r^u\rangle\top \\ & \leftrightarrow & \langle \Box p^u\rangle\Box(p \wedge q) \\ & \leftrightarrow & (\Box(p \wedge q))^p \\ & \leftrightarrow & \Box(p \to q). \end{array}$$

An information transition π *holds* in a context if the transition does not affect that context. For the example case, we can spell out the conditions for this as follows:

$$\begin{array}{l} s[\![\Box p^u; \Box(p \wedge q)?; \Box\neg r^u]\!]s \\ \text{ iff } \exists s' : s[\![\Box p^u]\!]s' \text{ and } s'[\![\Box(p \wedge q)?; \Box\neg r^u]\!]s \\ \text{ iff } s[\![\Box p^u]\!]s \text{ and } s[\![\Box(p \wedge q)?; \Box\neg r^u]\!]s \\ \text{ iff } s \models \Box p \text{ and } s \models \Box(p \wedge q) \text{ and } s \models \Box\neg r \\ \text{ iff } s \models \Box(p \wedge q \wedge \neg r). \end{array}$$

To end this section, note that the present perspective sheds an illuminating light on the phenomenon known as presupposition accommodation. Presupposition accommodation is the process performed by a benevolent audience in case an assertion is made with a presupposition which does not hold in the current context. In case the audience does not know that Bill is married and someone gossips that Bill's wife wants a divorce then the context is tacitly updated with the presupposition of that assertion as well. If we allow complex updates (by an obvious extension of the language), we can model this accommodation process as a shift from transition π to transition $(\langle\pi\rangle\top)^u; \pi$.

8 Embedded Presuppositions

Until now we have only considered presuppositions under sequential composition. If we assume presuppositions to have the form $\Box F?$, then a typical sequential composition of two updates under presupposition looks like this:

$$\Box F_1?; \Box G_1^u; \Box F_2?; \Box G_2^u.$$

The presupposition of this is given by:

$$\langle \Box F_1?; \Box G_1^u; \Box F_2?; \Box G_2^u\rangle\top.$$

This reduces to:

$$\Box F_1 \wedge \langle \Box G_1^u\rangle\langle \Box F_2?\rangle\top,$$

and further to:

$$\Box F_1 \wedge \Box(G_1 \to F_2),$$

with end result:

$$\Box(F_1 \wedge (G_1 \rightarrow F_2)).$$

Thus, we see that the boxed presupposition of the sequential composition of two updates is given by conjunction of the boxed presupposition of the first and the boxed implication of assertion of the first and presupposition of the second.

To consider presuppositions under negation, let us forget about 'downward' transitions π for the moment. (Note that we cannot define the negation of a downdate with α as the assertion that downdating with α itself would lead to inconsistency, for if a downdate with α is possible at all, it will never lead to inconsistency. Also, the negation of a downdate with α cannot be construed as the assertion that downdating with α is impossible, for the impossibility of a downdate with α just means that α is a logical truth.)

An *update transition* is a transition π with the property that $s[\![\pi]\!]s'$ implies $s \sqsubseteq s'$. The *presupposition* of an update transition is the set $\{s \in S \mid \exists s' \sqsupseteq s : s[\![\pi]\!]s'\}$. This set is characterized by $\langle\pi\rangle\top$. The *content* of an update transition is the set $\{s \in S \mid s[\![\pi]\!]s\}$. This set is characterized by fix (π).

Negating an update α^u can be construed as updating with the assertion that making update α itself would yield inconsistency. Thus, we can stipulate:

$$\neg(\alpha^u) = ([\alpha^u]\Box\bot)^u.$$

If we define $\varphi_1^u \Rightarrow \varphi_2^u$ as $\neg(\varphi_1^u; \neg\varphi_2^u)$ (update implication) and $\varphi_1^u \sqcup \varphi_2^u$ as $\neg(\neg\varphi_1^u; \neg\varphi_2^u)$ (update disjunction), then we can easily derive:

- $[\![\neg(\Box F^u)]\!] = [\![\Box\neg F^u]\!]$,
- $[\![\Box F^u \Rightarrow \Box G^u]\!] = [\![\Box(F \rightarrow G)^u]\!]$,
- $[\![\Box F^u \sqcup \Box G^u]\!] = [\![\Box(F \vee G)^u]\!]$.

In the general case where an update transition π may have a presupposition, we have two options: either the negation preserves the presupposition or it cancels it. We will explore the first option. Suppose π is an upward transition. Then we define $\neg\pi$ as follows:

$$\neg\pi \stackrel{\text{def}}{=} \langle\pi\rangle\top?; ([\pi]\Box\bot)^u.$$

Thus, $\neg\pi$ has the same presupposition as π, but it updates to the minimal state(s) where updating with π would yield inconsistency.

As regards the second option, an obvious choice for a definition of negated updating which cancels presuppositions is $([\pi]\Box\bot)^u$. This is an update to a state where doing π itself would lead to inconsistency. Now suppose π has a presupposition, let us say $\Box p?$, and assume $\Diamond\neg p$ holds in the current state. Then $([\pi]\Box\bot)^u$ would loop in the current state, showing that $([\pi]\Box\bot)^u$ does not have $\Box p?$ as presupposition.

If we spell out the semantics for $\neg\pi$ we get this:

$$[\![\neg\pi]\!] = A - B,$$

where

$$\begin{aligned} A \;=\; & \{\langle s, s'\rangle \in S \times S \mid s \sqsubseteq s', s'[\![\pi]\!]\langle\emptyset, w\rangle \text{ (for some } w\text{)}, \\ & \text{and for all } s'' \text{ with } s \sqsubseteq s'' \sqsubseteq s' \;\&\; s''[\![\pi]\!]\langle\emptyset, w\rangle \text{ (for some } w\text{)} \\ & \hfill s' \sqsubseteq s''\}, \end{aligned}$$

and

$$B = \{\langle s, s'\rangle \mid s \sqsubseteq s', s \models [\pi]\bot\}.$$

Note that the earlier stipulation for $\neg(\alpha^u)$ is a special case of this.

We can now define dynamic implication and dynamic disjunction for the general case of update transitions π_1 and π_2. To calculate what happens to presuppositions under 'dynamic implication', we can make use of the fact that

$$\neg(\Box F?; \Box G^u) = \Box F?; \Box\neg G^u$$

and of the fact that for all π:

$$\langle\neg\pi\rangle\top \leftrightarrow \langle\pi\rangle\top.$$

This is just a reflection of the fact that $\neg\pi$ has the same presupposition as π.

Here is the calculation for presupposition under dynamic implication:

$$\begin{aligned}
&\langle (\Box F_1?; \Box G_1^u) \Rightarrow (\Box F_2?; \Box G_2^u)\rangle\top \\
&\leftrightarrow \langle \neg((\Box F_1?; \Box G_1^u); \neg(\Box F_2?; \Box G_2^u))\rangle\top \\
&\leftrightarrow \langle \neg(\Box F_1?; \Box G_1^u; \Box F_2?; \Box\neg G_2^u)\rangle\top \\
&\leftrightarrow \langle \Box F_1?; \Box G_1^u; \Box F_2?; \Box\neg G_2^u\rangle\top \\
&\leftrightarrow \Box F_1 \wedge \langle \Box G_1^u\rangle \Box F_2 \\
&\leftrightarrow \Box(F_1 \wedge (G_1 \rightarrow F_2)).
\end{aligned}$$

For presupposition under 'dynamic disjunction' we get:

$$\begin{aligned}
&\langle (\Box F_1?; \Box G_1^u) \sqcup (\Box F_2?; \Box G_2^u)\rangle\top \\
&\leftrightarrow \langle \neg(\neg(\Box F_1?; \Box G_1^u); \neg(\Box F_2?; \Box G_2^u))\rangle\top \\
&\leftrightarrow \langle \neg(\Box F_1?; \Box\neg G_1^u; \Box F_2?; \Box\neg G_2^u)\rangle\top \\
&\leftrightarrow \langle \Box F_1?; \Box\neg G_1^u; \Box F_2?; \Box\neg G_2^u\rangle\top \\
&\leftrightarrow \Box F_1 \wedge \langle \Box\neg G_1^u\rangle \Box F_2 \\
&\leftrightarrow \Box(F_1 \wedge (G_1 \vee F_2)).
\end{aligned}$$

Thus, calculating presuppositions of complex updates in terms of assertions and presuppositions of their components gives the following table:

update procedure	presupposition
$(\Box F_1?; \Box G_1^u); (\Box F_2?; \Box G_1^u)$	$\Box(F_1 \wedge (G_1 \rightarrow F_2))$.
$\neg(\Box F?; \Box G^u)$	$\Box F$.
$(\Box F_1?; \Box G_1^u) \Rightarrow (\Box F_2?; \Box G_1^u)$	$\Box(F_1 \wedge (G_1 \rightarrow F_2))$.
$(\Box F_1?; \Box G_1^u) \sqcup (\Box F_2?; \Box G_1^u)$	$\Box(F_1 \wedge (G_1 \vee F_2))$.

This is a *boxed* version of Karttunen's table of presupposition projection for 'and', 'not', 'if then' and 'or'.

9 Digression: Error States

The treatment of presupposition failure in terms of error states of van Eijck 1993 van Eijck 1994 is motivated by an obvious parallel between presupposition failure in natural language and error abortion in imperative programming. Consider the program statement (7).

(7) `x := y/z`

If at the point of execution of this statement register z happens to contain the value 0 then execution will be aborted with an error statement like 'Floating point error: division by zero attempted'.

(8) `IF z <> 0 THEN x := y/z`

In the statement (8) the dangerous case of $z = 0$ is tested for in the program code, and the danger of error abortion is staved off.

This suggests analyzing presupposition failure as 'moving to an error state'. Taking error abortion into account in the semantics of deterministic imperative programming boils down to changing the semantic interpretation function for program statements into a partial function: error abortion is the case where there is no next state.

The epistemic state of a program always consists of the current memory state, so it turns out that error abortion analysis arises as a special case of the present epistemic analysis, where there are just two state sets: $\langle\{w\}, w\rangle$ (the consistent state) and $\langle\emptyset, w\rangle$ (the inconsistent state). Thus we get:

Success case $w(p) = 1, w(q) = 1$:

$$\langle\{w\}, w\rangle [\![\Box p?; \Box q^u]\!] \langle\{w\}, w\rangle$$

Failure case $w(p) = 1, w(q) = 0$:

$$\langle\{w\}, w\rangle [\![\Box p?; \Box q^u]\!] \langle\emptyset, w\rangle$$

Error abortion case $w(p) = 0$:

$$\langle\{w\}, w\rangle[\![\Box p?; \Box q^u]\!] \text{ ERROR} .$$

In nondeterministic imperative programming, program statements are interpreted as relations. Taking error abortion into account here means changing the interpretation relation into a partial relation. Executing a program statement π in state s now gives three possibilities: (1) there are proper next states (and maybe the program can also make a transition to 'error'), (2) there are no next states, and (3) the program can only make a transition to the error state. Again, an error state semantics for dynamic predicate logic (Groenendijk and Stokhof 1991) in the style of van Eijck 1993 turns out to be a special case of the present epistemic analysis, where there is just one first order model around, and where the possible states are the assignment functions over this single model (intuitively, the states encode the interpretations for indefinite noun phrases that are 'still in the running'). In this set-up, an update with presupposition for 'the king of France is bald' could be rendered as $\Box\exists!xKx?; \Box(x :=?; Kx; Bx)^u$. This gets us into the topic of the next section.

10 Presupposition and Quantification

As an example of a presupposition of quantified expressions, we look at the case of uniqueness presuppositions of singular definite descriptions. Dynamic versions of predicate logic have been proposed to deal with growth of information about anaphoric possibilities of a piece of natural language text. The most important ones of these are file change semantics (Heim 1982), discourse representation theory (Kamp 1981), and dynamic predicate logic (Groenendijk and Stokhof 1991). This kind of dynamics can be, but need not be, combined with the dynamics of information updating using predicate logical formulas. Here we will concentrate on 'epistemic dynamics' for purposes of exposition, and sketch a system of information updating for standard predicate logic.

To model information growth in predicate logic, the simplest possible set-up confines attention to one particular predicate logical model M for the language under consideration, and then uses sets of variable assignments for that model as information states. Thus, if $M = \langle \text{dom}\,(M), \text{int}\,(M)\rangle$ is given, and if V is the set of variables for the predicate logical language under consideration, then $A = \text{dom}\,(M)^V$ is the set of assignments, and $S = \{\langle i, a\rangle \mid i \subseteq A, a \in A\}$ is the set of information states. The relation $\sqsubseteq$ on S is given by $s \sqsubseteq s'$ iff $s = \langle i, a\rangle, s' = \langle j, a\rangle$ and $i \supseteq j$. Absurd information states are states of the form $\langle\emptyset, a\rangle$.

Fix a language L: let a set of individual constants C and a set of predicate constants P_n (where n denotes the arity of the constant) be given. Assume V is a set of individual variables. Assume $c \in C$, $v \in V$, $R \in P_n$.

$$\begin{array}{rcl} t & ::= & c \mid v \\ \varphi & ::= & \bot \mid Rt_1 \cdots t_n \mid t_1 = t_2 \mid \neg\varphi \mid (\varphi_1 \wedge \varphi_2) \mid \exists v\varphi. \end{array}$$

Let L_1 be the language that allows epistemic statements over L:

$$\psi \quad ::= \quad \varphi \mid \Diamond\varphi \mid \Box\varphi.$$

The Tarskian satisfaction relation $M \models_a \varphi$ is defined in the usual manner. In terms of this we define an interpretation for L_1 (i.e., a specification function σ) as follows:

$$\begin{array}{rcl} \sigma(\varphi) & = & \{s \in S \mid s = \langle i, a\rangle \text{ and } M \models_a \varphi\}, \\ \sigma(\Diamond\varphi) & = & \{\langle i, a\rangle \in S \mid \exists a' \in i \text{ with } M \models_{a'} \varphi\}, \\ \sigma(\Box\varphi) & = & \{\langle i, a\rangle \in S \mid \forall a' \in i : M \models_{a'} \varphi\}. \end{array}$$

The dynamically extended language now becomes:

$$\begin{array}{rcl} t & ::= & c \mid v \\ \varphi & ::= & \bot \mid Rt_1 \cdots t_n \mid t_1 = t_2 \mid \neg\varphi \mid (\varphi_1 \wedge \varphi_2) \mid \exists v\varphi \\ \psi & ::= & \varphi \mid \Diamond\varphi \mid \Box\varphi \mid \text{dom}\,(\pi) \mid \text{ran}\,(\pi) \mid \text{fix}\,(\pi). \\ \pi & ::= & \psi^u \mid \psi? \mid (\pi_1; \pi_2). \end{array}$$

Minimal updates are defined as before. The definitions of the dynamic operators are the same as before. Again, presuppositions are given by dom (π) and assertions by fix (π). The distinction between presupposition failure and updating with a piece of information inconsistent with the current information state is given by 'no further transition possible' versus 'transition to an absurd state'.

(9) The king of France is eating a frog.

An update with the information expressed by (9) is expressed in this format as (10).

(10) $\Box\exists!xKx?; \Box\exists x(Kx \wedge \exists y(Fy \wedge Exy))^u$.

The presupposition is given by:

(11) $\langle\Box\exists!xKx?; \Box\exists x(Kx \wedge \exists y(Fy \wedge Exy))^u\rangle\top$.

This is equivalent to:

(12) $\Box\exists!xKx$.

The assertion is given by:

(13) $\text{fix}\,(\Box\exists!xKx?; \Box\exists x(Kx \wedge \exists y(Fy \wedge Exy))^u)$.

This is equivalent to:

(14) $\Box(\exists!xKx \wedge \exists x(Kx \wedge \exists y(Fy \wedge Exy)))$.

Note that because our epistemic states are based on a single first order model the epistemic operators $\Diamond$ and $\Box$ are not very expressive. They serve to make the distinction between being able to make an update to an absurd state (uttering a falsehood) and not being able to make a further transition at all (error abortion). Indeed, since possible worlds are variable assignments, if F is a predicate logical formula without free variables, then the difference between $\Box F$ and $\Diamond F$ shows up only in absurd information states.

Of course, the epistemic modalities become more expressive once we redefine our information states in terms of *sets* of first order models.

11 Conclusion

We have sketched a system of epistemic dynamic logic to model presupposition and presupposition failure. Lots of logical questions remain to be answered. For instance: is the logic of propositional up- and downdating decidable? We conjecture that it is. What does a complete axiomatisation of this logic look like? What are the properties of the systems one gets by imposing further conditions on the information structures? What do the obvious variations on the combination of presupposition and quantification look like? The simplest variation is to replace standard predicate logic by dynamic predicate logic. This yields the dynamic error state semantics of van Eijck 1993. Another variation is to replace states based on single first order models by states based on sets of models. This gives an epistemic first order update logic. Finally, we can combine the two in various ways (see Eijck and Cepparello 1994 and Groenendijk et al. 1994). In all cases, the main thing is to get at the right definition of the information structure $\langle S, \sqsubseteq\rangle$. Jaspars and Krahmer 1995 provide a very useful starting point for this in the form of an overview of current systems of dynamic logic from the perspective of information structures.

Acknowledgements

I would like to thank Jan Jaspars for some illuminating conversations on the topic of information updating, and Makoto Kanazawa for his very helpful comments on two earlier versions of this paper.

References

Beaver, D.I. 1992. The Kinematics of Presupposition. In *Proceedings of the Eighth Amsterdam Colloquium*, ed. P. Dekker and M. Stokhof, 17–36. ILLC, University of Amsterdam.

Benthem, J. van. 1991. Logic and the Flow of Information. Technical Report LP-91-10. ILLC, University of Amsterdam.

Chellas, B.F. 1980. *Modal Logic: An Introduction.* Cambridge University Press.

Eijck, J. van. 1993. The Dynamics of Description. *Journal of Semantics* 10:239–267.

Eijck, J. van. 1994. Presupposition Failure — A Comedy of Errors. *Formal Aspects of Computing* 6A:766–787.

Eijck, J. van, and G. Cepparello. 1994. Dynamic Modal Predicate Logic. In *Dynamics, Polarity and Quantification*, ed. M. Kanazawa and C.J. Piñón. 251–276. CSLI, Stanford.

Eijck, J. van, and F.J. de Vries. 1995. Reasoning About Update Logic. *Journal of Philosophical Logic* 24:19–45.

Groenendijk, J., and M. Stokhof. 1991. Dynamic Predicate Logic. *Linguistics and Philosophy* 14:39–100.

Groenendijk, J., M. Stokhof, and F. Veltman. June, 1994. Coreference and Modality. Manuscript, ILLC, Amsterdam.

Halpern, J., and Y. Moses. 1985. Towards a Theory of Knowledge and Ignorance: Preliminary Report. In *Logics and Models of Concurrent Systems*, ed. K.R. Apt. 459–476. Springer.

Heim, I. 1982. *The Semantics of Definite and Indefinite Noun Phrases.* Doctoral dissertation, University of Massachusetts, Amherst.

Heim, I. 1983. On the Projection Problem for Presuppositions. *Proceedings of the West Coast Conference on Formal Linguistics* 2:114–126.

Heim, I. 1992. Presupposition Projection and the Semantics of the Attitude Verbs. *Journal of Semantics* 9(3):183–221. Special Issue: Presupposition, Part 1.

Jaspars, J. to appear. Partial Up and Down Logic. *Notre Dame Journal of Formal Logic.*

Jaspars, J., and E. Krahmer. 1995. Unified Dynamics. Manuscript, CWI, Amsterdam.

Kamp, H. 1981. A Theory of Truth and Semantic Representation. In *Formal Methods in the Study of Language*, ed. J. Groenendijk et al. Mathematisch Centrum, Amsterdam.

Kanazawa, M. 1994. Completeness and Decidability of the Mixed Style of Inference with Composition. In *Proceedings 9th Amsterdam Colloquium*, ed. P. Dekker and M. Stokhof. 377–390. ILLC, Amsterdam.

Karttunen, L. 1973. Presuppositions of Compound Sentences. *Linguistic Inquiry* 4:169–193.

Karttunen, L. 1974. Presupposition and Linguistic Context. *Theoretical Linguistics* 181–194.

Karttunen, L., and S. Peters. 1979. Conventional Implicature. In *Syntax and Semantics 11: Presupposition*, ed. C.-K. Oh and D. Dinneen. 1–56. Academic Press.

Krahmer, E. 1994. Partiality and dynamics; theory and application. In *Proceedings 9th Amsterdam Colloquium*, ed. P. Dekker and M. Stokhof. 391–410. ILLC, Amsterdam.

Moore, R.C. 1984. Possible world semantics for autoepistemic logic. In *Proceedings AAAI Workshop on Non-Monotonic Reasoning*, 344–354. New Paltz, NY.

Rijke, M. de. 1994. Meeting Some Neighbours. In *Logic and Information Flow*, ed. J. van Eijck and A. Visser. 170–195. MIT Press, Cambridge, Mass.

Stalnaker, R. 1972. Pragmatics. In *Semantics of Natural Language*, ed. D. Davidson and G. Harman. 380–397. Reidel.

Stalnaker, R. 1974. Pragmatic Presuppositions. In *Semantics and Philosophy*, ed. M.K. Munitz and P.K. Unger. 197–213. New York University Press.

Veltman, F. 1991. Defaults in Update Semantics. Technical report. Department of Philosophy, University of Amsterdam. To appear in the *Journal of Philosophical Logic.*

Zeevat, H. 1992. Presupposition and Accommodation in Update Semantics. *Journal of Semantics* 9(4):379–412. Special Issue: Presupposition, Part 2.